Recurrent Neural Networks (Part - 2)

Sumit Chopra
Facebook

Recap

Standard RNNs

Training: Backpropagation Through Time (BPTT)

Application to sequence modeling

Language modeling

Applications: Automatic speech recognition, Machine translation

Main problems in training

Major Shortcomings

Handling of complex non-linear interactions

Difficulties using BPTT to capture long-term dependencies

Exploding gradients

Vanishing gradients

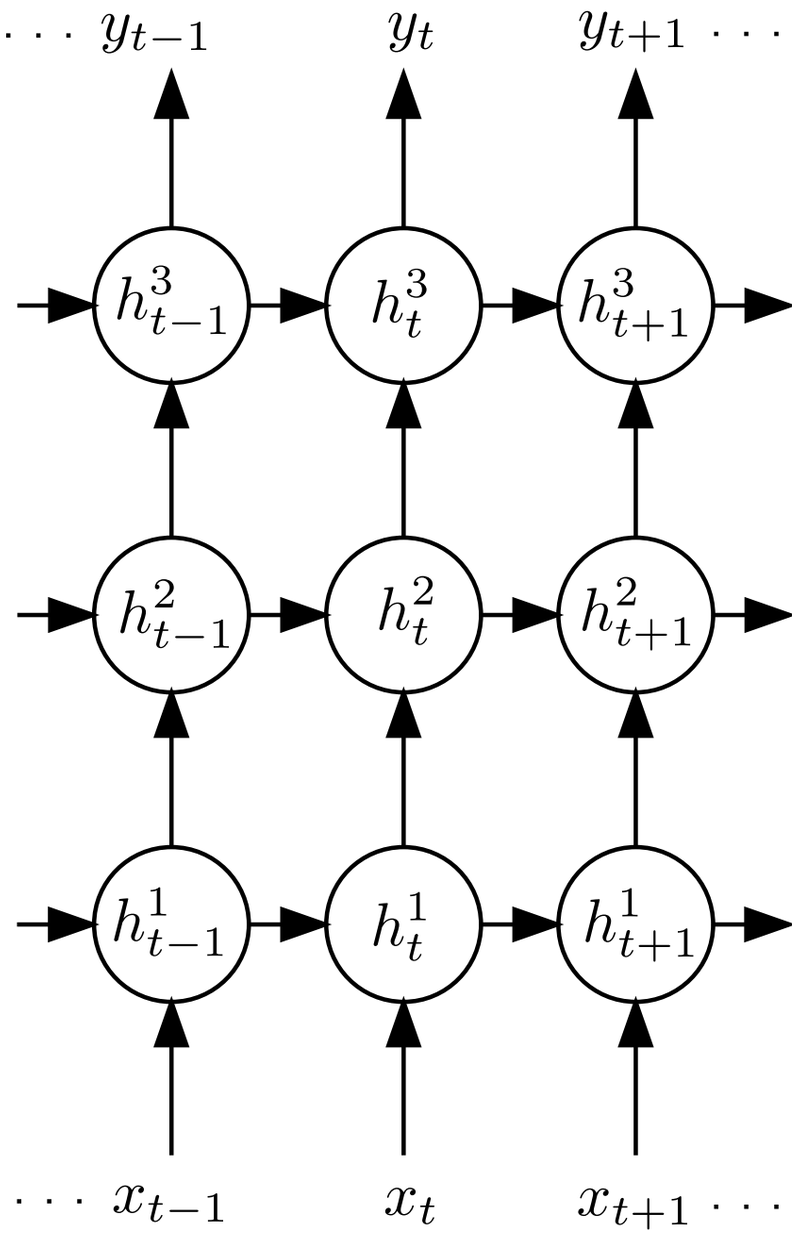
Handling Non-Linear Interactions

Handling Non-Linear Interactions

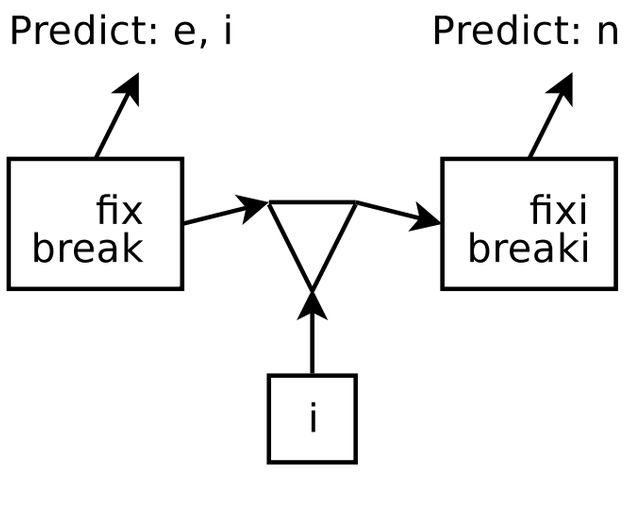
have depth not only in temporal dimension

but also in space (at each time step)

empirically shown to provide significant improvement in tasks like ASR, Un-supervised training using videos

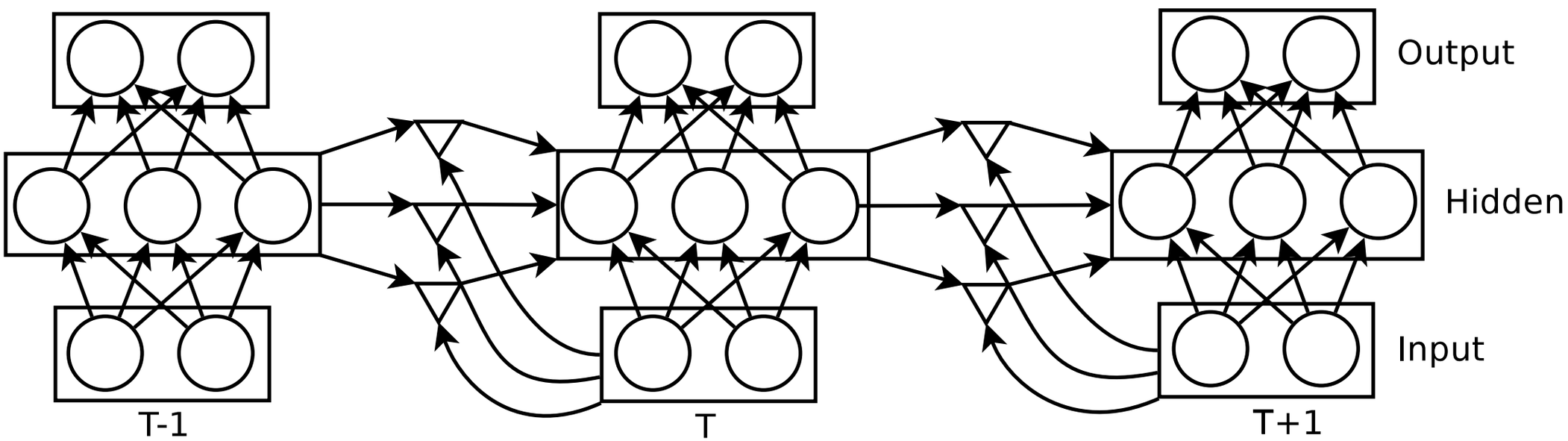


Handling Non-Linear Interactions



Gated RNNs

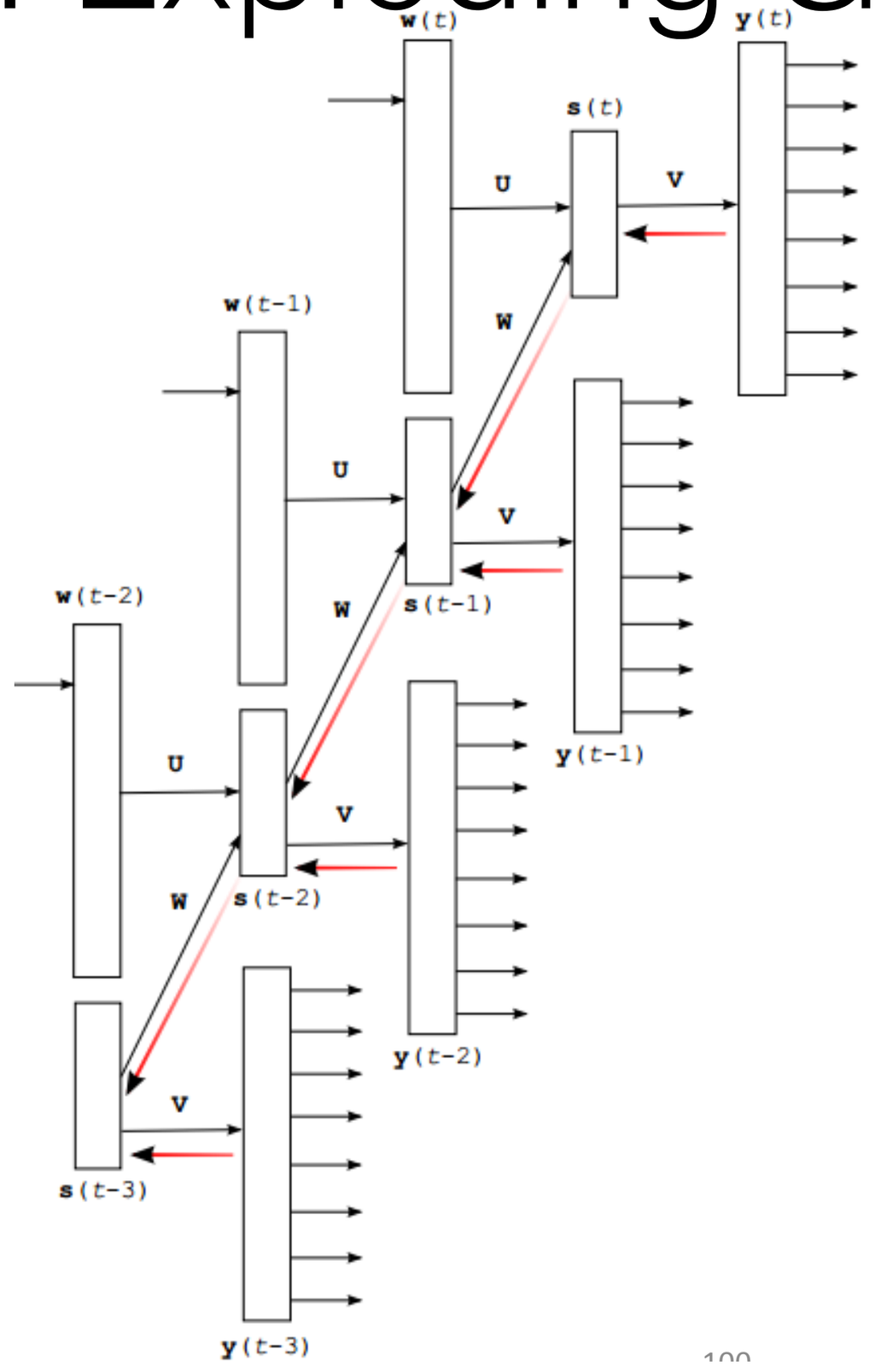
shown to work on character based language modeling



Sutskever et.al., 2011:Generating Test with Recurrent Networks

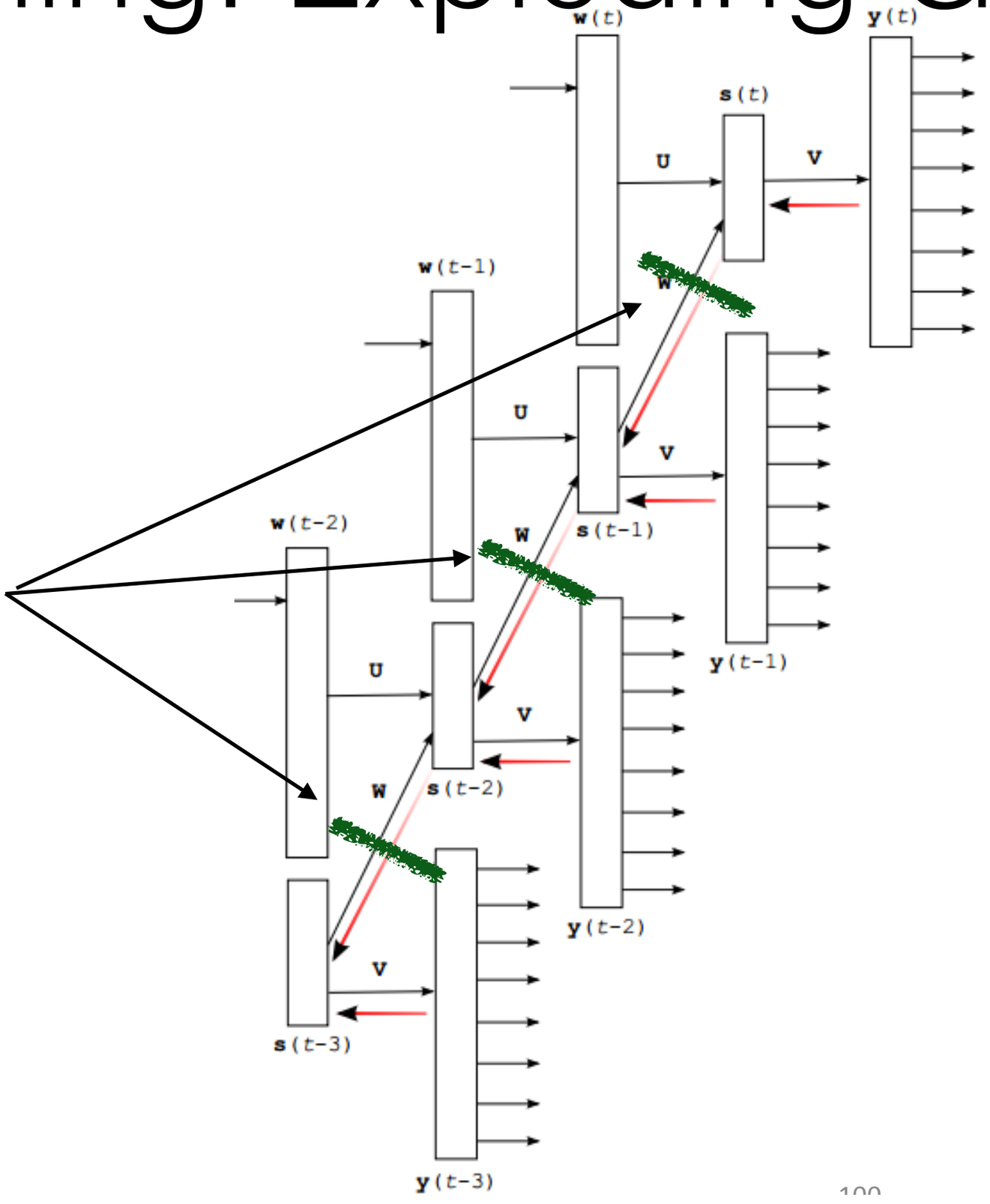
Training: Exploding Gradients

Gradient Clipping during BPTT



Training: Exploding Gradients

Gradient Clipping during BPTT



Training: Vanishing Gradients

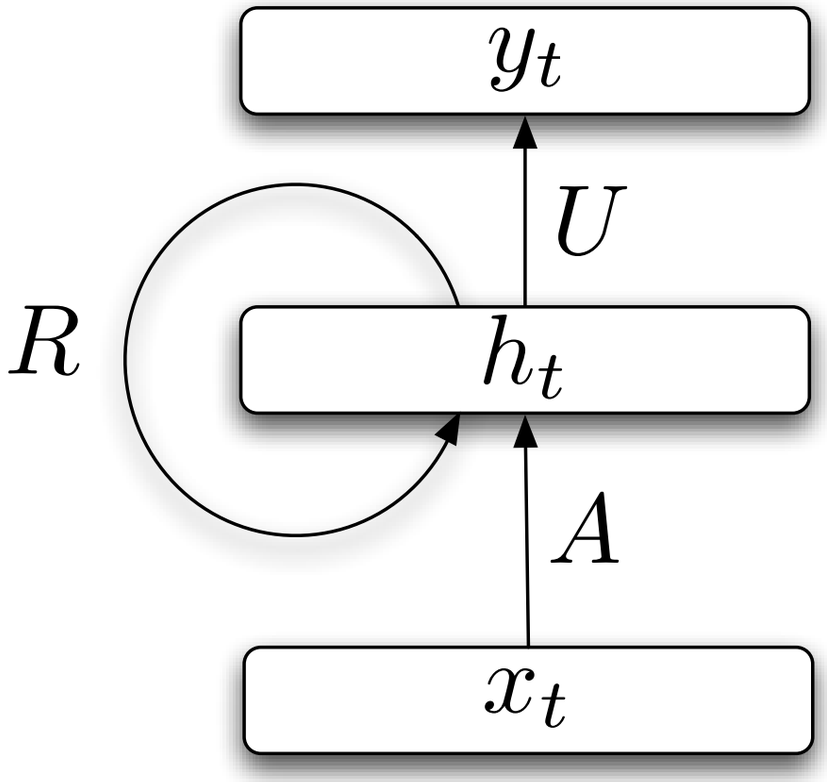
Multiple schools of thought

better initialization of the recurrent matrix and using momentum during training

Sutskever et.al.,: On The Importance of Initialization and Momentum in Deep Learning

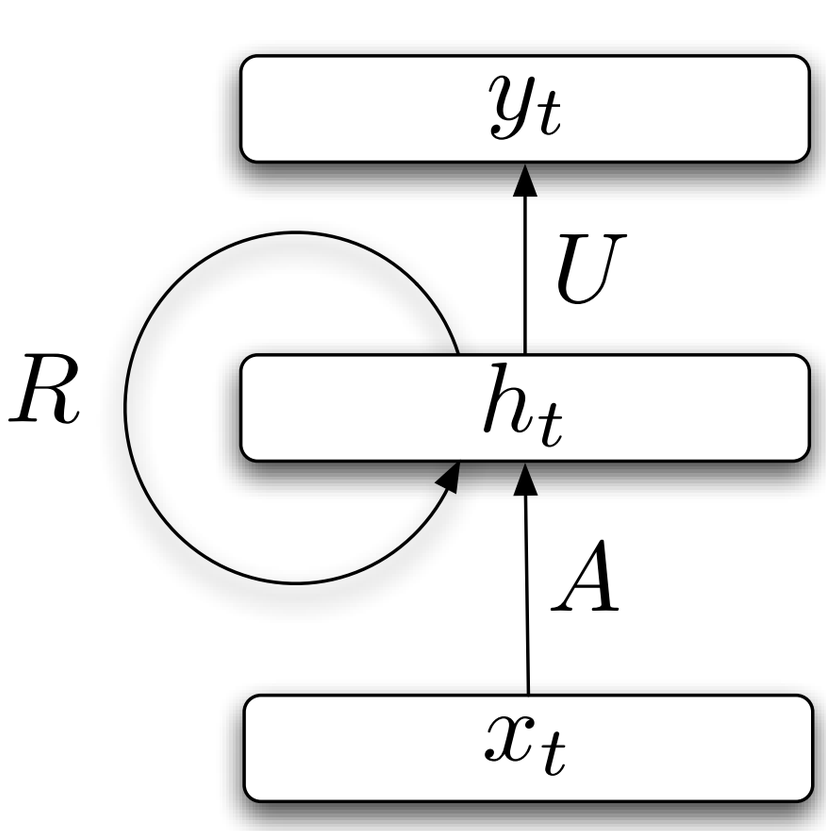
modifying the architecture

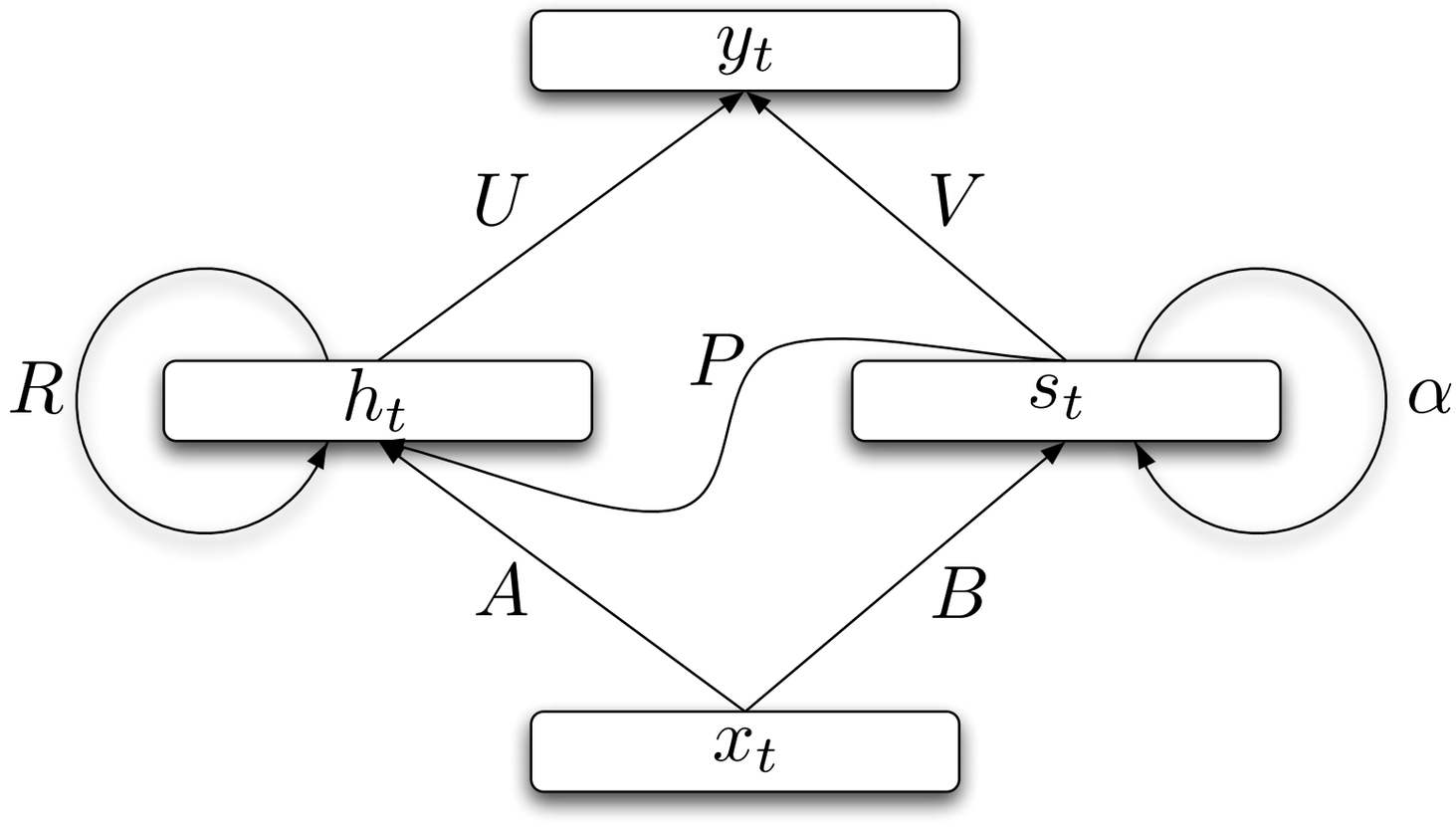
Structurally Constrained RNNs



Mikolov et.al., 2015:Learning Longer Memory in Recurrent Neural Networks

Structurally Constrained RNNs





$$
\begin{aligned}
s_t &= (1-\alpha)Bx_t + \alpha s_{t-1}, \\
h_t &= \sigma\left(Ps_t + Ax_t + Rh_{t-1}\right), \\
y_t &= f\left(Uh_t + Vs_t\right)
\end{aligned}
$$

Structurally Constrained RNNs

Language Modeling on Penntree Bank Corpus

Model	#hidden	#context	Test Perplexity
Ngram	-	-	141
Ngram + cache	-	-	125
SRN	50	-	144
SRN	100	-	129
SRN	300	-	129
SCRN	100	40	**115**
SCRN	300	40	**115**

Structurally Constrained RNNs

Language Modeling on Text8 Corpus

Model	#hidden	context = 0	context = 40	context = 80
SCRN	100	245	189	184
SCRN	300	202	165	164
SCRN	500	184	162	**161**

Long Short-Term Memory (LSTM)

recently gained a lot of popularity

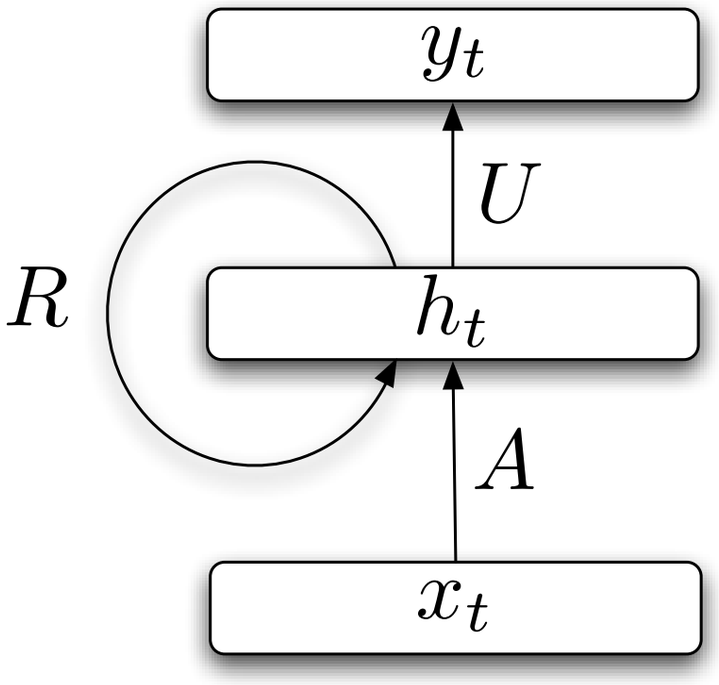
have explicit memory “cells” to store short-term activations

the presence of additional gates partly alleviates the vanishing gradient problem

multi-layer versions shown to work quite well on tasks which have“medium term” dependencies

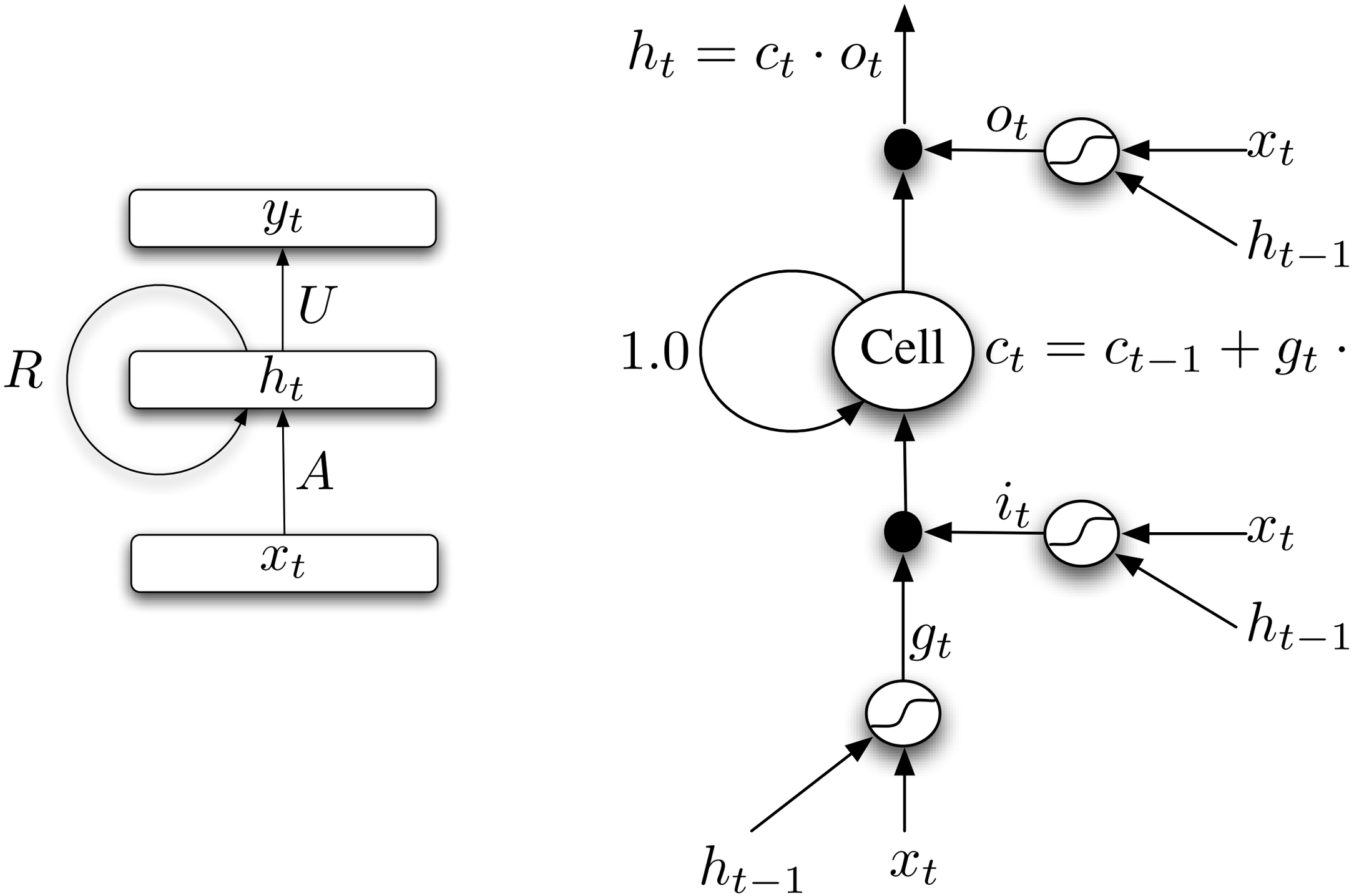
Hochreiter et.al., 1997: Long Short-Term Memory

Long Short-Term Memory (LSTM)



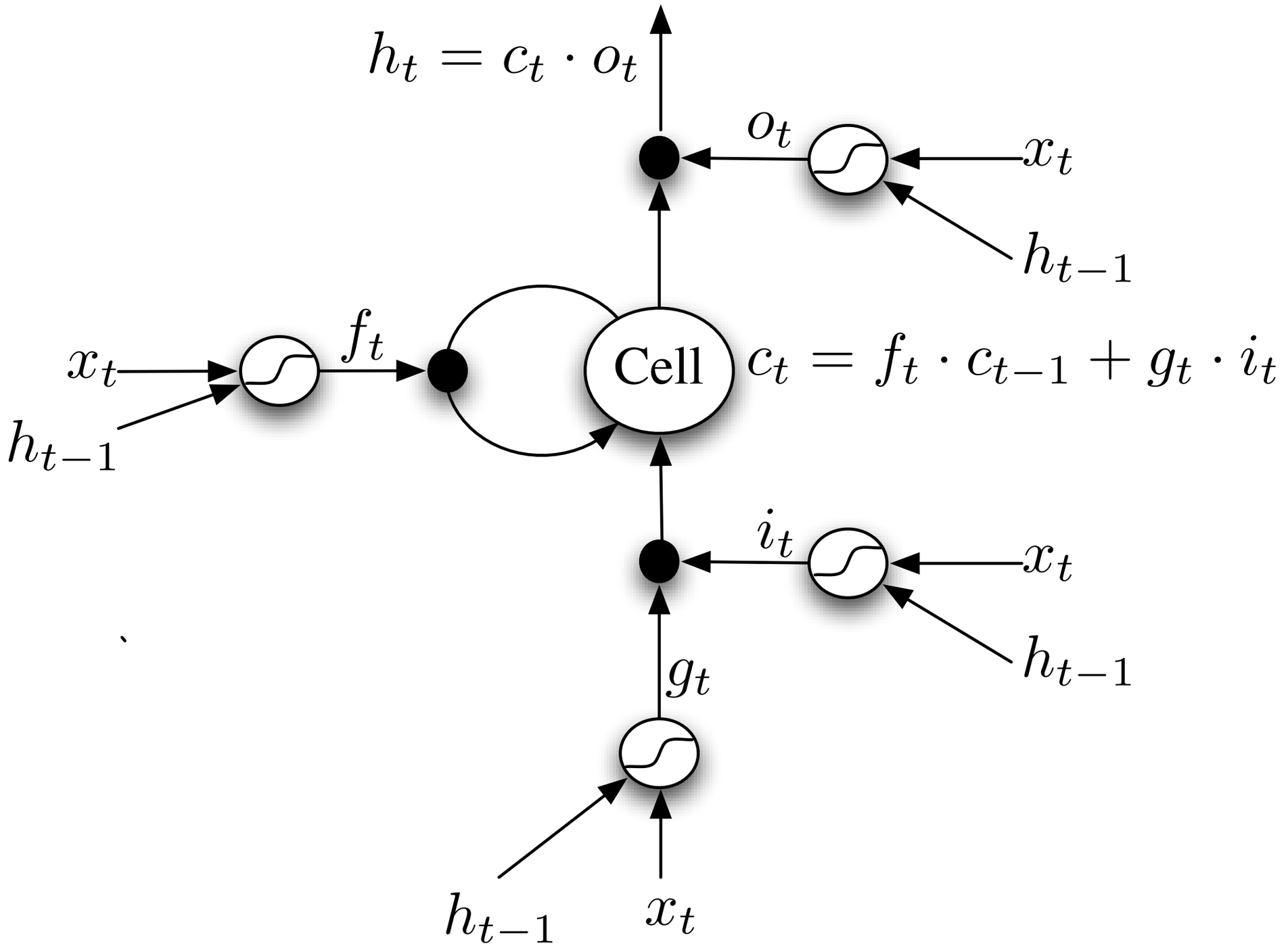
Hochreiter et.al., 1997: Long Short-Term Memory

Long Short-Term Memory (LSTM)



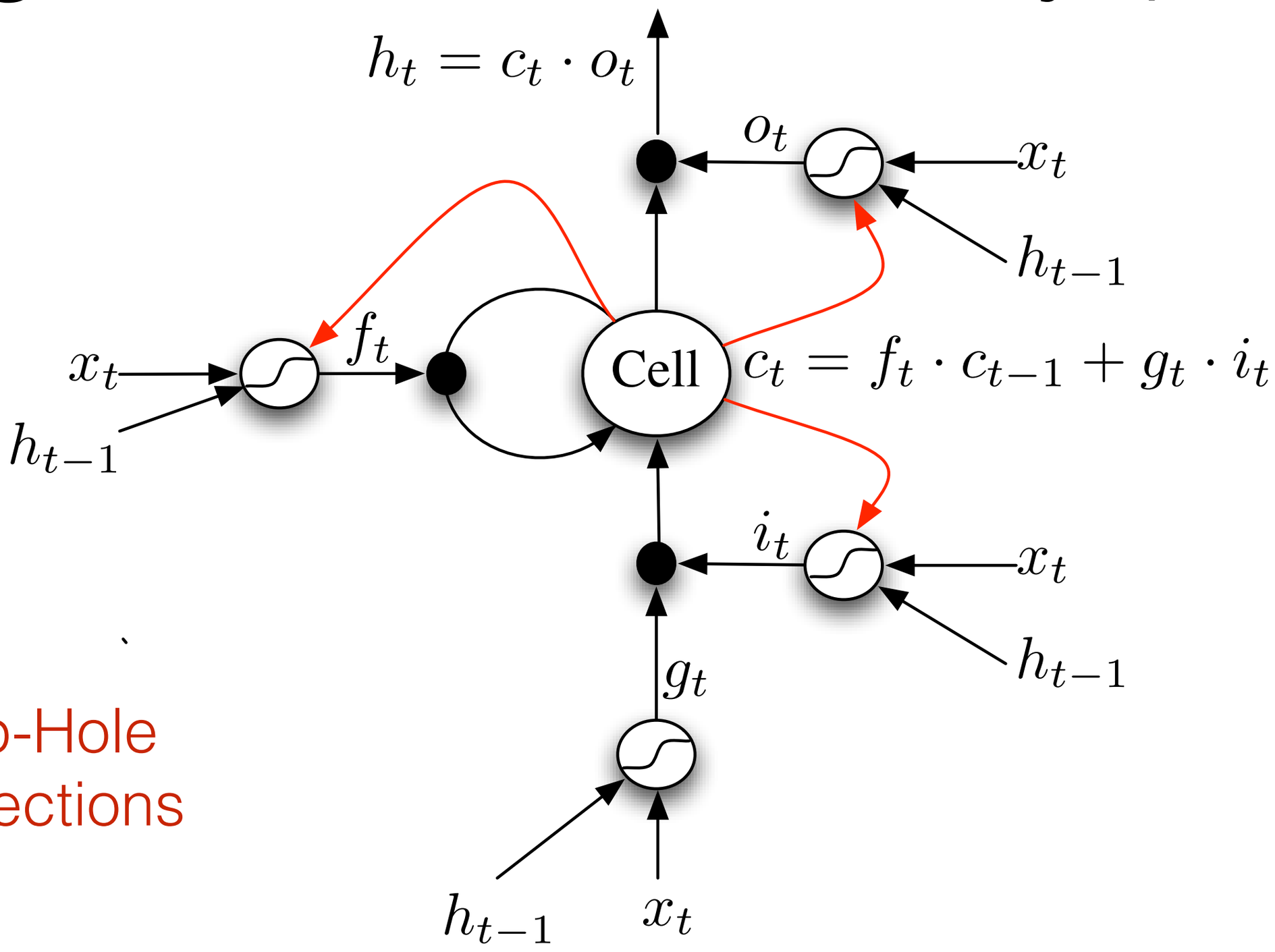
Hochreiter et.al., 1997: Long Short-Term Memory

Long Short-Term Memory (LSTM)



Hochreiter et.al., 1997: Long Short-Term Memory

Long Short-Term Memory (LSTM)

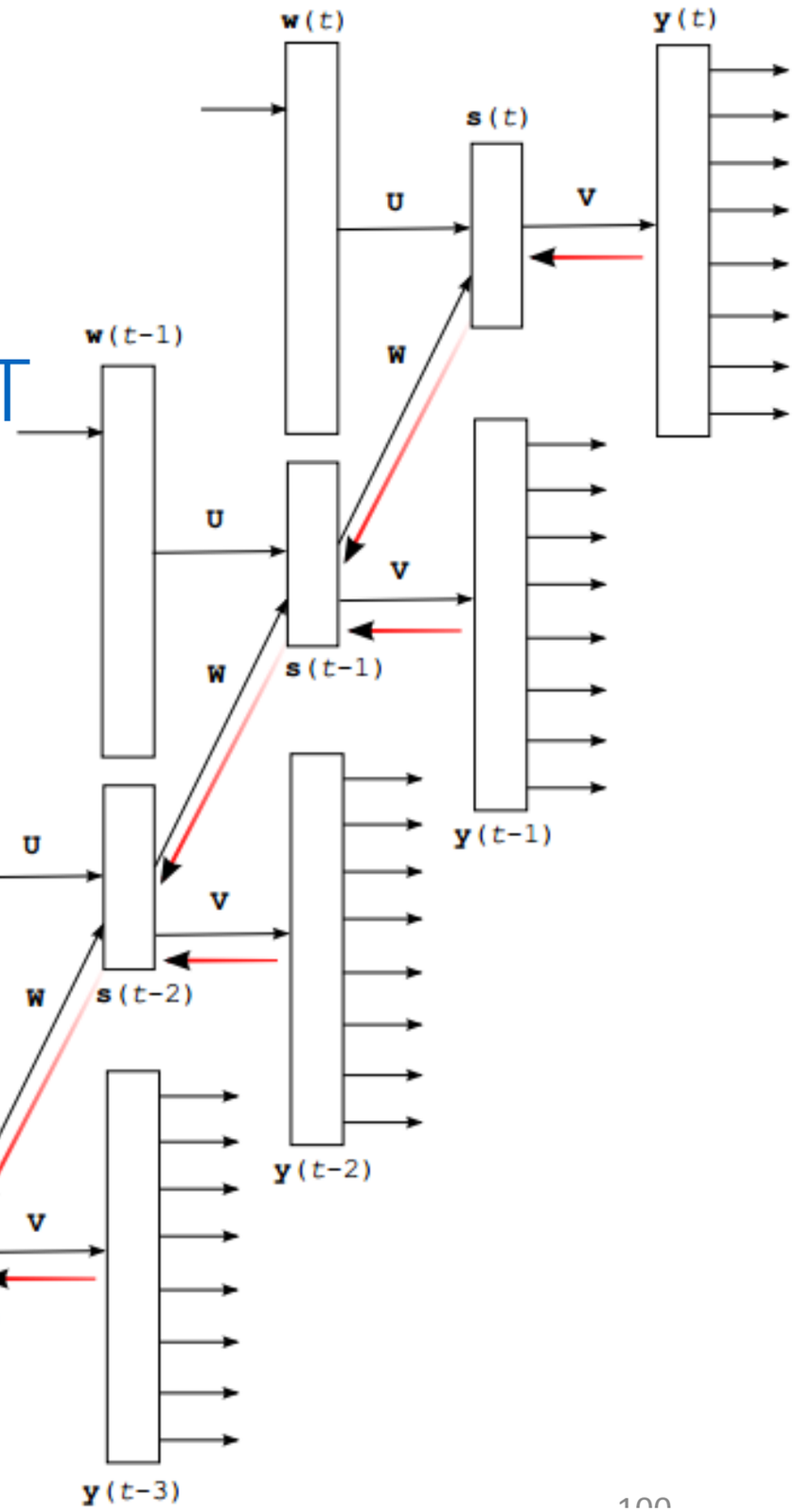


Peep-Hole Connections

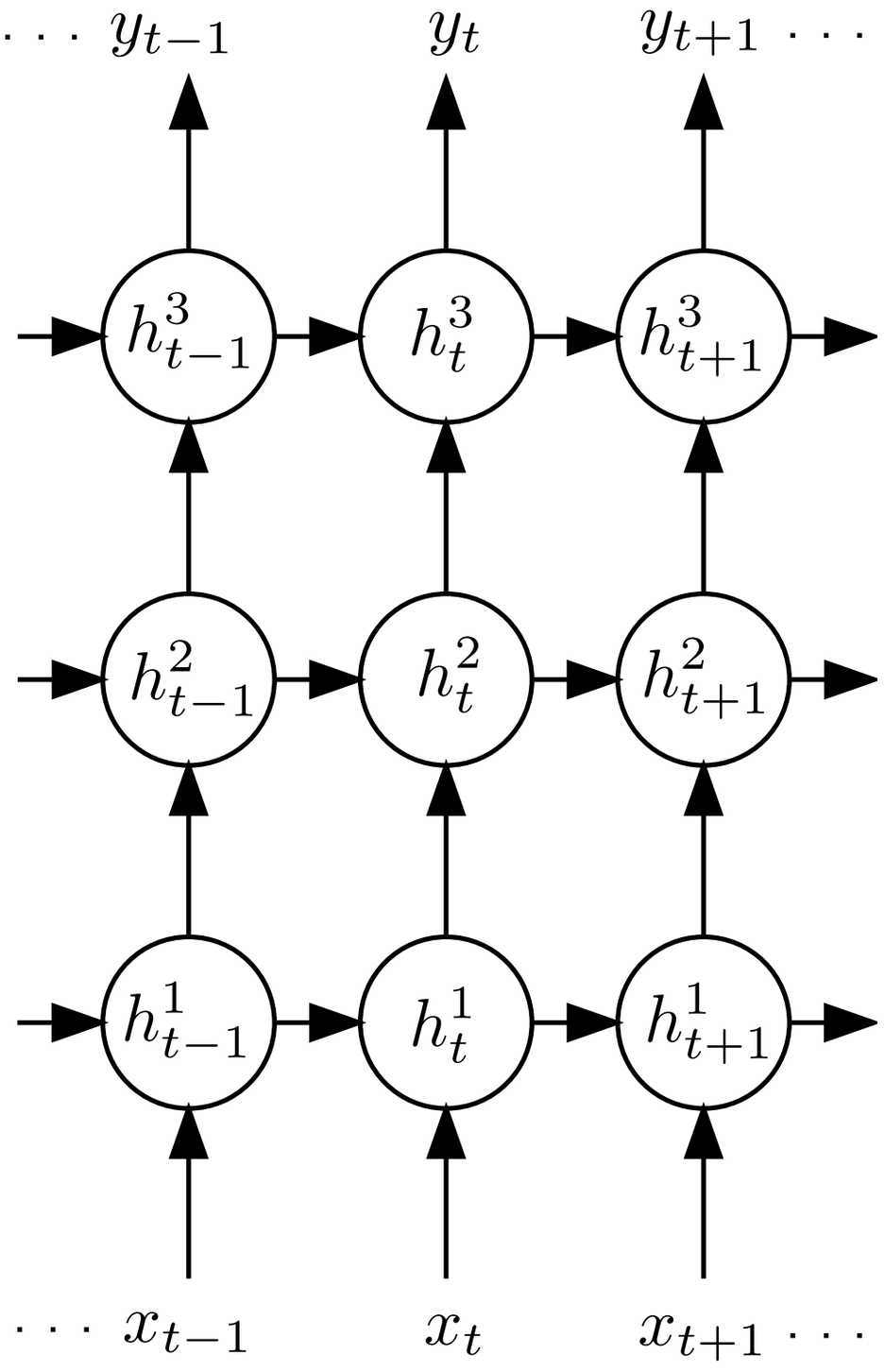
Hochreiter et.al., 1997: Long Short-Term Memory

LSTM Training

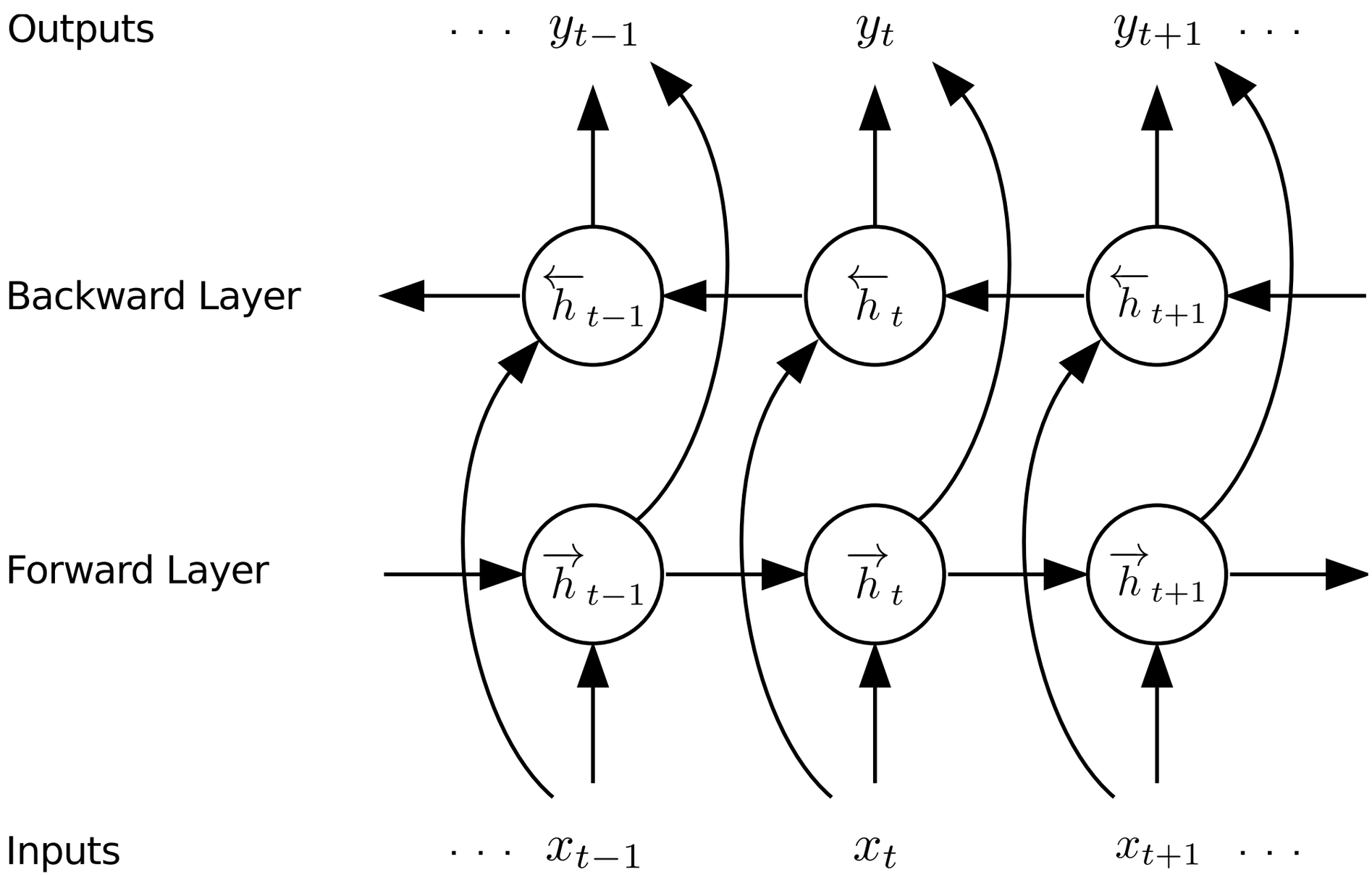
Backpropagation Through Time: BPTT



Deep LSTMs


$\cdots$ y_{t-1}
y_t
y_{t+1} $\cdots$
h^3_{t-1}
h^3_t
h^3_{t+1}
h^2_{t-1}
h^2_t
h^2_{t+1}
h^1_{t-1}
h^1_t
h^1_{t+1}
$\cdots$ x_{t-1}
x_t
x_{t+1} $\cdots$

Bi-Directional LSTMs

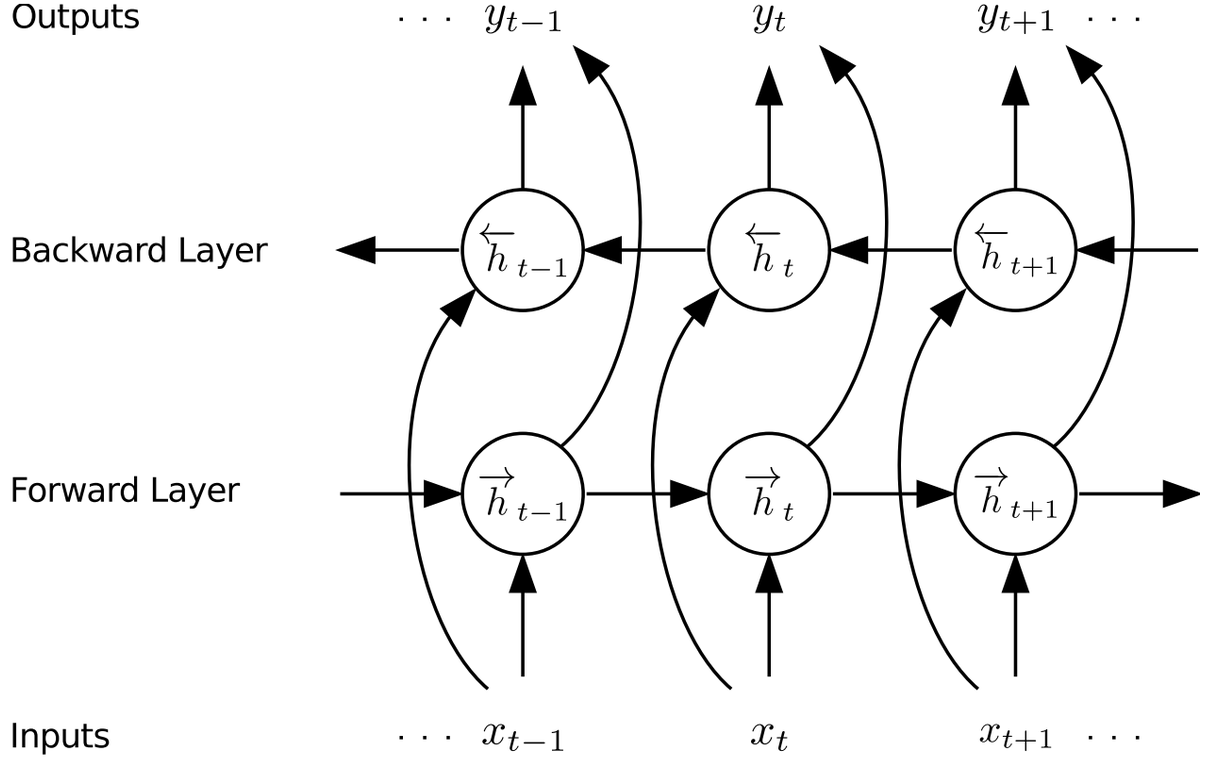


Applications of LSTMs

Automatic Speech Recognition

Use bi-directional LSTMs to represent the audio sequence

plug a classifier on top of the representation to directly predict phone classes



Graves et. al., 2014: Speech Recognition with Deep Recurrent Neural Networks

Automatic Speech Recognition

Table 1. TIMIT Phoneme Recognition Results. 'Epochs' is the number of passes through the training set before convergence. 'PER' is the phoneme error rate on the core test set.

NETWORK	WEIGHTS	EPOCHS	PER
CTC-3L-500H-TANH	3.7M	107	37.6%
CTC-1L-250H	0.8M	82	23.9%
CTC-1L-622H	3.8M	87	23.0%
CTC-2L-250H	2.3M	55	21.0%
CTC-3L-421H-UNI	3.8M	115	19.6%
CTC-3L-250H	3.8M	124	18.6%
CTC-5L-250H	6.8M	150	18.4%
TRANS-3L-250H	4.3M	112	18.3%
PRETRANS-3L-250H	**4.3M**	**144**	**17.7%**

Graves et. al., 2014: Speech Recognition with Deep Recurrent Neural Networks

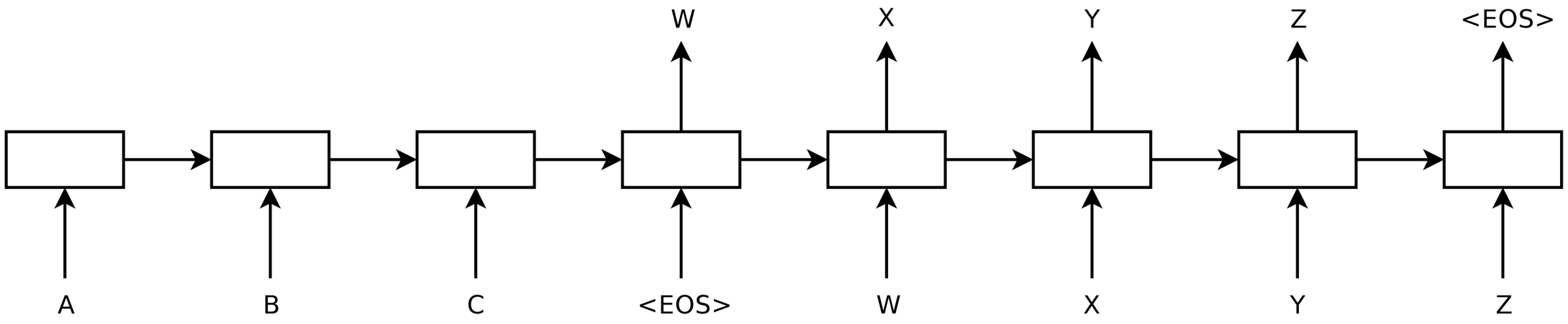
Sequence to Sequence Learning

A B C —> W X Y Z

Machine Translation

Short Text Response Generation

Sentence Summarization



Sutskever at. al., 2014: Sequence to Sequence Learning with Neural Network

Sequence to Sequence Learning

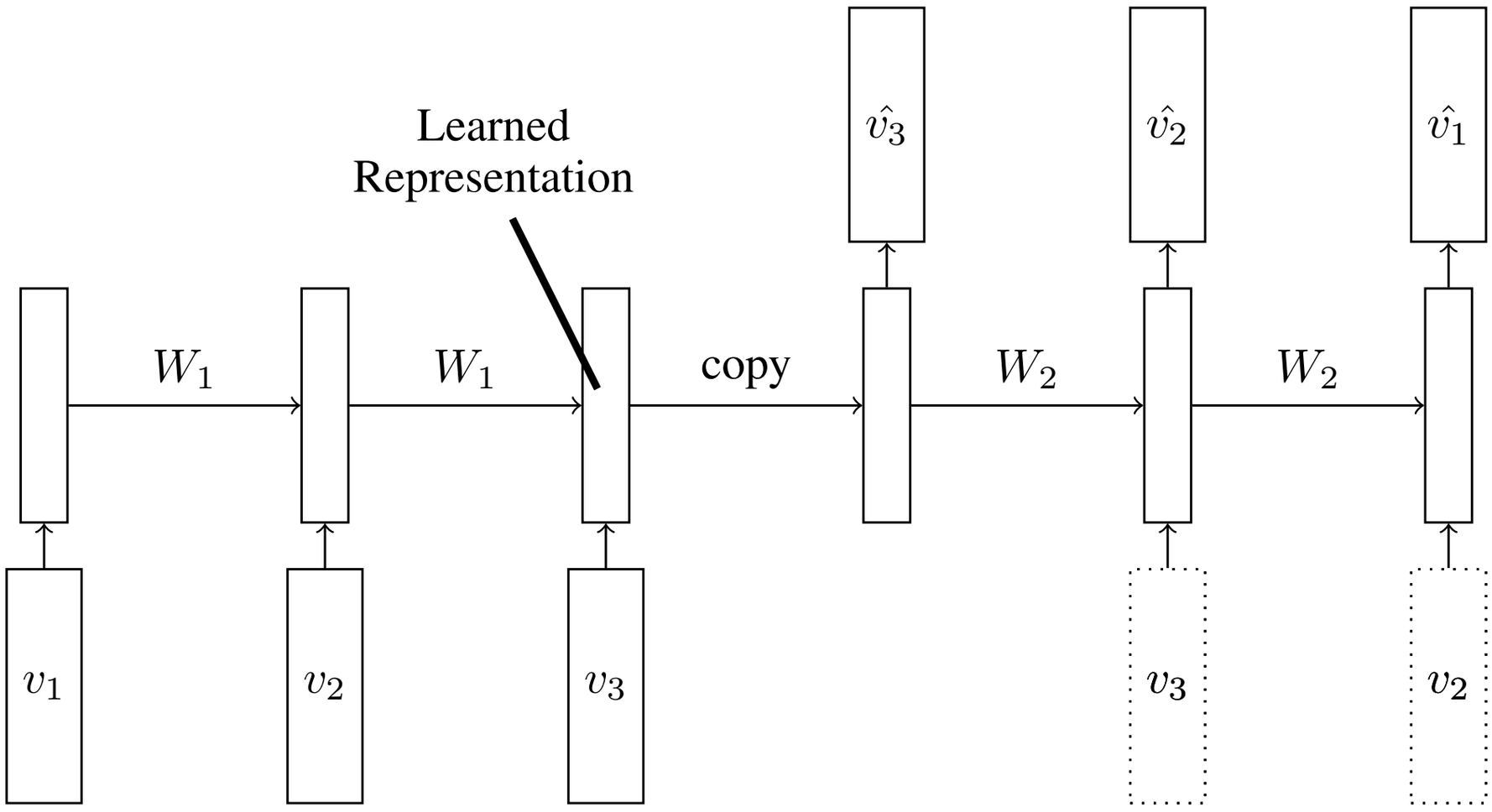
Method	test BLEU score (ntst14)
Bahdanau et al. [2]	28.45
Baseline System [29]	33.30
Single forward LSTM, beam size 12	26.17
Single reversed LSTM, beam size 12	30.59
Ensemble of 5 reversed LSTMs, beam size 1	33.00
Ensemble of 2 reversed LSTMs, beam size 12	33.27
Ensemble of 5 reversed LSTMs, beam size 2	34.50
Ensemble of 5 reversed LSTMs, beam size 12	**34.81**

State-of-the-art WMT'14 result: 37.0

Sutskever at. al., 2014: Sequence to Sequence Learning with Neural Network

Unsupervised Training on Video

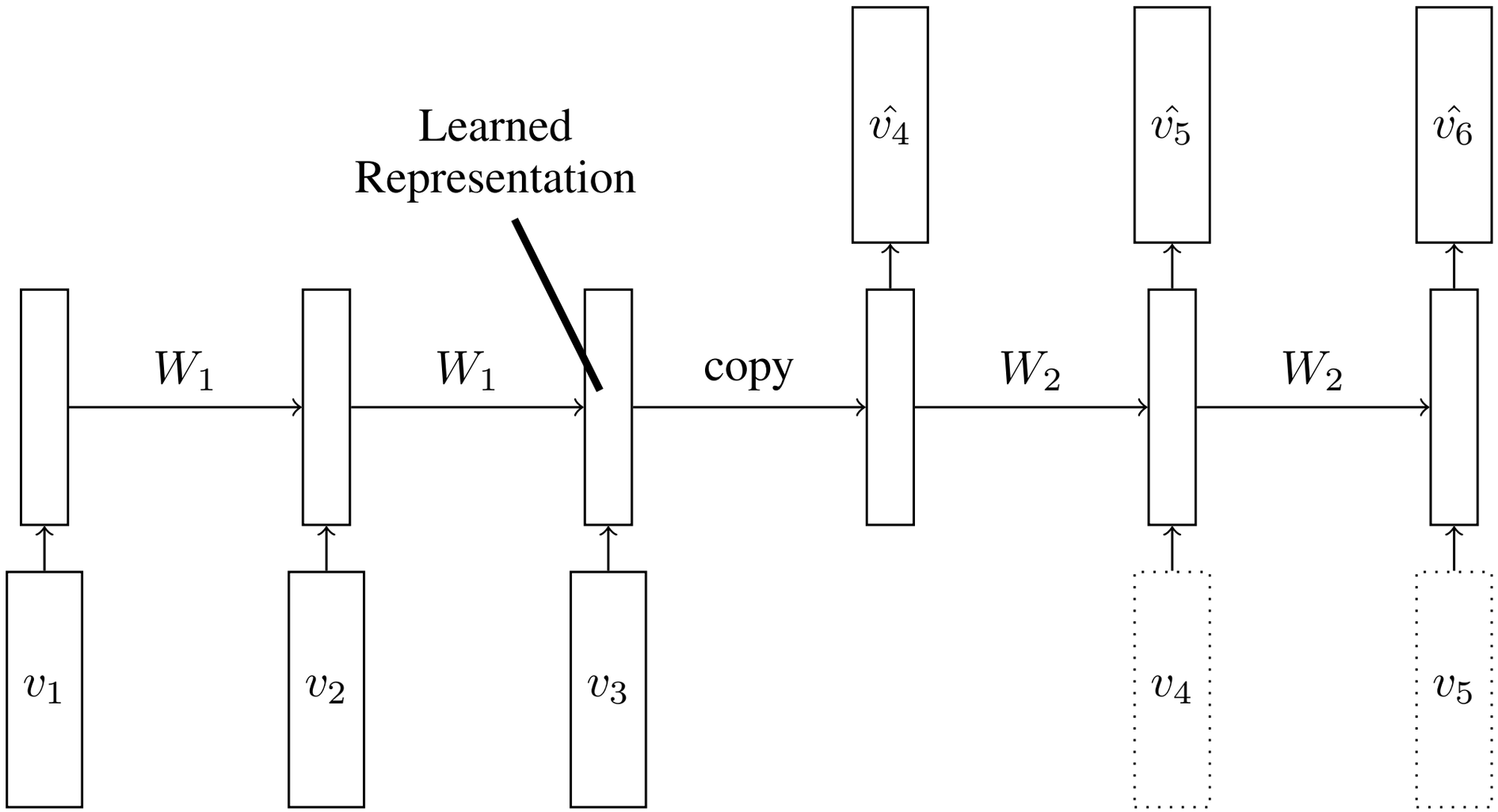
Auto-encoder Model



Srivastav et. al., 2014: Unsupervised Learning of Video Representation using LSTMs

Unsupervised Training on Video

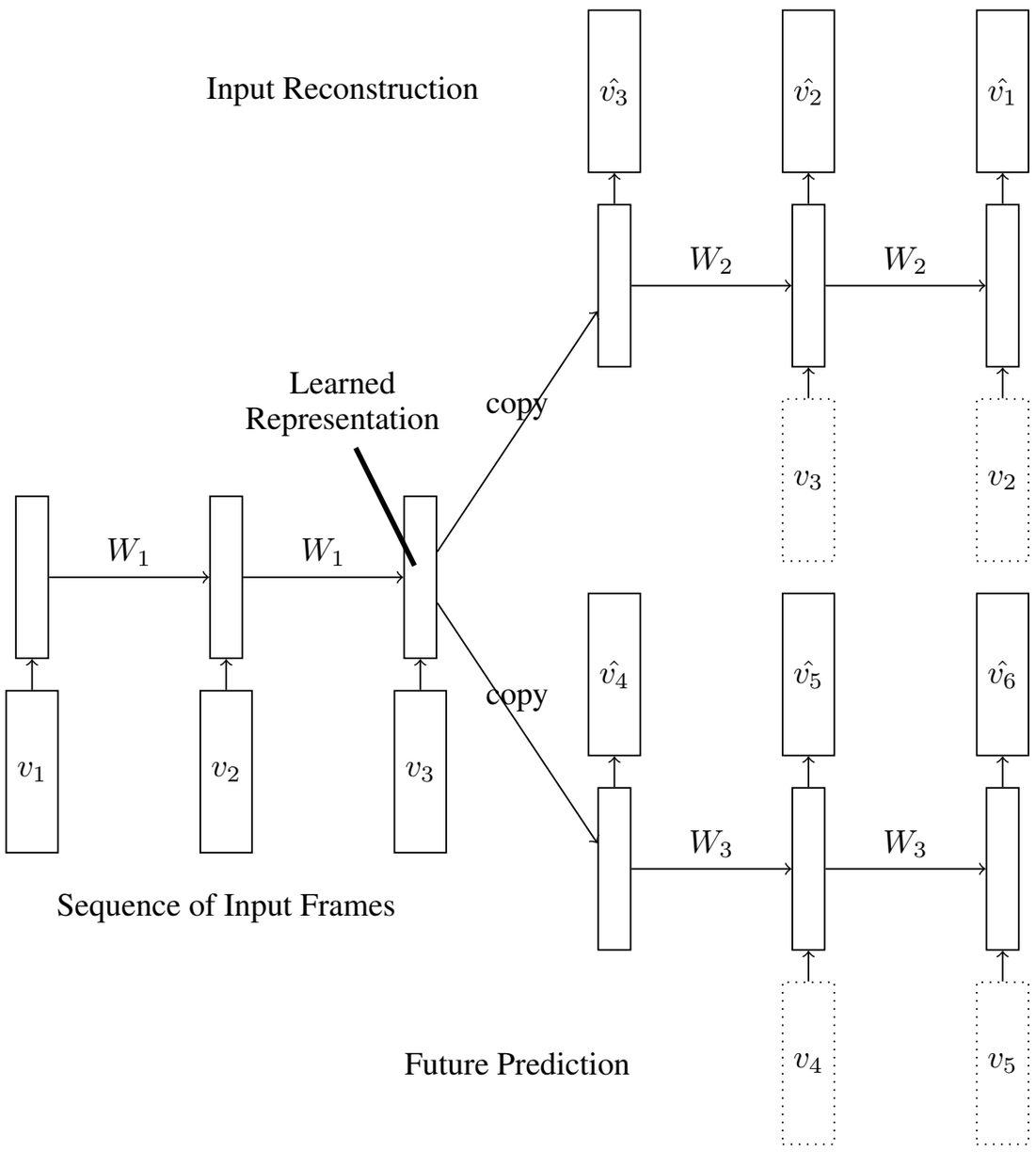
Future Frame Predictor Model



Srivastav et. al., 2014: Unsupervised Learning of Video Representation using LSTMs

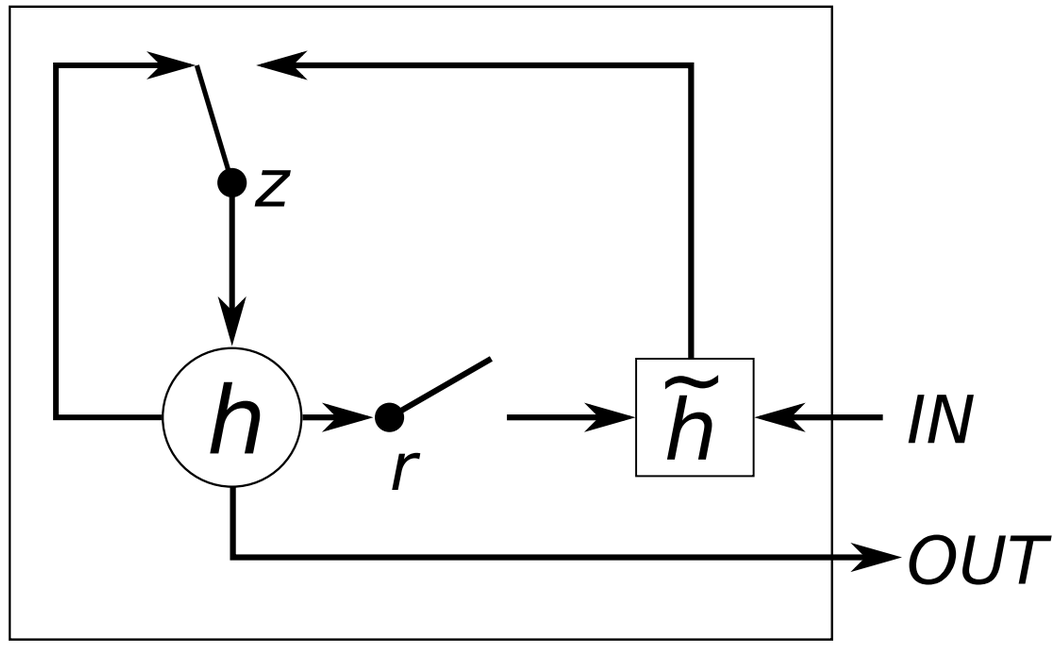
Unsupervised Training on Video

Composite Model



Srivastav et. al., 2014: Unsupervised Learning of Video Representation using LSTMs

Gated Recurrent Units



Update gate: $z_t^j = \sigma\left(W_z \mathbf{x}_t + U_z \mathbf{h}_{t-1}\right)^j .$

Reset gate: $r_t^j = \sigma\left(W_r \mathbf{x}_t + U_r \mathbf{h}_{t-1}\right)^j .$

Candidate activation: $\tilde{h}_t^j = \tanh\left(W \mathbf{x}_t + U\left(\mathbf{r}_t \odot \mathbf{h}_{t-1}\right)\right)^j ,$

$$h_t^j = (1 - z_t^j) h_{t-1}^j + z_t^j \tilde{h}_t^j ,$$

Implementation

Torch code available (soon!)

Standard RNN

LSTMs

SCRNN

and other models..

GPU compatible

Open Problems

Encoding long-term memory into RNNs

Speed-up the RNN training

Control problems

Language understanding